Homework 2 Solution

CS6923 Machine Learning

Part I

1. (a) **(1 point)** That is $\frac{0}{0+3} = 0$

   (b) **(1 point)** $\frac{1}{1+2} = \frac{1}{3}$

   (c) **(1 point)** $\frac{56+41}{56+2+1+41} = \frac{97}{100}$

2. **(2 point)**

From the question, we know that for the given training example (x1, x2) we assign it to C_2 if $g_2(x_1, x_2) >= g_1(x_1, x_2)$

Otherwise, we assign it to C_1.

So $g(x_1, x_2) = g_1(x_1, x_2) - g_2(x_1, x_2) = 8x_2 + x_1 + 2$

3. **(2 point)**

P(spam | x_1, x_2) = $\frac{1}{1+e^{-(6-6+1)}} = \frac{e}{1+e}$

P(not spam | x_1, x_2) = 1 - P(spam | x_1, x_2) = $\frac{1}{1+e}$

Since

Expected cost of classifying [x_1,x_2] as not spam = (Cost of False Negative)*P(spam|x_1,x_2) = 5*P(spam|x_1,x_2)

Expected cost of classifying [x_1,x_2] as spam = (Cost of False Positive)*P(not spam|x_1,x_2) = 2*P(not spam|x_1,x_2)

and $5 * \frac{e}{1+e} > 2 * \frac{1}{1+e}$

So the prediction that is spam has smaller risk.

4. (a) **(1 point)**

$$bias = E\left[\frac{\sum_{x \in \chi} x}{N+1}\right] - \mu = \frac{N\mu}{N+1} - \mu = -\frac{\mu}{N+1}$$

   (b) **(1 point)**

No, the answer will not change.

5. (a) **(2 points)**

$$\hat{\mu}_+ = \left[\mu_1^+, \mu_2^+\right] = [1.833,\ 3.2]^T$$

$$\hat{\mu}_- = \left[\mu_1^-, \mu_2^-\right] = [1.5,\ 2.533]^T$$

$$\sum_+ = \begin{bmatrix} 2.5356 & 3.9333 \\ 3.9333 & 6.14 \end{bmatrix}$$

$$\sum_- = \begin{bmatrix} 0.42 & 0.99 \\ 0.99 & 2.4422 \end{bmatrix}$$

(b) **(1 point)**

$$\log p = \log\left(\frac{1}{(2\pi)^{d/2}\left|\sum\right|^{1/2}} e^{-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)}\right) = -\frac{1}{2}(x-\mu)^T\sum^{-1}(x-\mu) - \frac{d}{2}\log(2\pi) - \frac{1}{2}\left|\sum\right|$$

$$\log p(\hat{x}|+) = -3.835 - \log(2\pi)$$

$$\log p(\hat{x}|-) = -1.047 - \log(2\pi)$$

$$\log p(\hat{x}|-) > \log p(\hat{x}|+), \text{ so the ML hypothesis is '}-\text{'}.$$

(c) **(2 points)**

Reasons it might be a good idea:

1. Fewer parameters to estimate means less chance of overfitting.
2. This allows you to use all the training data to compute the parameters of the single covariance matrix, which could be an advantage if you don't have much training data (since otherwise you can only use positive examples to estimate the covariance matrix for the positive class, and similarly for the negative class).
3. (Another reason: leads to a linear discriminant function, rather than a quadratic one)

Reasons it might not be a good idea:

1. If the actual covariance matrices for the two classes are very different from each other, it doesn't make a lot of sense to try to use the same estimated covariance matrix for both of them. You get inaccurate models for the two distributions, potentially leading to bad classification performance.

6. (a) **(1 point)**

$$g(x) = 0.0757x - 1.976$$

(b) **(1 point)**

$$g(475) = 33.9899$$

7. (a) **(1 point)**

x1	x2	label

0.68	0.83	+
1.00	1.00	+
0.00	0.00	+
0.24	0.08	-
0.46	0.52	-
0.63	0.81	-

(b) **(1 point)**

x is scaled to $\begin{bmatrix} 1.024 \\ 0.096 \end{bmatrix}$

The Euclidean distances between x and each training data are
[0.808, 0.904, 1.029, 0.785, 0.705, 0.814].

Since k=1 in this case, the nearest neighbor is 0.705. Therefore, the predicted label is '-'.

Part II

a

(i) **(1 point)**

Predicted label : 1

(ii) **(1 point)**

		Predicted	
		1	0
Correct	1	TP:209	FN:64
	0	FP:134	TN:93

(iii) **(1 point)**

Accuracy: 60.4%

True Positive Rate: 0.765

False Positive Rate: 0.590

(iv) **(1 point)**

Predicted label : 1

(v) **(1 point)**

		Predicted	
		1	0
Correct	1	TP:212	FN:61
	0	FP:136	TN:91

(vi) + (vii) **(1 point)**
True Positive Rate: 0.777
False Positive Rate: 0.599
Accuracy: 60.6%
(viii) **(1 point)**

		Predicted	
		1	0
Correct	1	TP:273	FN:0
	0	FP:227	TN:0

(b) **(1 point)**
Under the distance function we are using here, long documents in the training set are more likely than short ones to be nearest neighbors of test documents. e.g., Suppose that A and B are training documents where A is short and B is long.
If test document C is long, then B and C are likely to have more words in common than A and C, just because B and C both have lots of words in them (in fact, the number of words that A and C have in common can be no more than the number of words in A).
And if test document C is short, then B still has an advantage over A, because the long length of C gives more chances for the words in A to appear. As a result, short documents in the training set might often be "ignored" in making the k-NN decisions.

c
(i)
For k = 3:
Cross-validation accuracy: 66%
For k = 7:
Cross-validation accuracy: 65.8%
For k = 99:
Cross-validation accuracy: 61.27%
(ii)

		Predicted	
		1	0
Correct	1	TP:212	FN:61
	0	FP:144	TN:83

Accuracy: 59%

(d) (i) and (ii) **(2 points)**
There are many possible answers to these questions.
You received full credit if you did something that seemed like it might be effective, you explained clearly how your distance function was computed, and you explained clearly why it seemed like a good idea.

(Note: You did not get any points off if your new distance function performed poorly in the experiments.)